IN THE CLAIMS:

The following listing of claims will replace all prior versions, and listings, of claims in the application.

1. (Currently Amended) A graphics system comprising:

a set of graphics accelerators, wherein each [[card]] of the graphics accelerators comprises a rendering processor, an internal frame buffer, and a video data port; and

a series of filtering units, wherein each of the filtering units couples to a video data port of a corresponding one of the graphics accelerators;

wherein each of the graphics accelerators is configured to: (a) generate a stream of samples in response to received graphics primitives, (b) add a corresponding dither value to the color components of the samples to obtain dithered color components, (c) buffer the dithered color components in the internal frame buffer, and (d) forward truncated versions of the dithered color components to a corresponding filtering unit; and

wherein for a specific pixel, each of the filtering units is configured to compute corresponding partial sums from the truncated versions of the dithered color components.

2. (Original) The graphics system of claim 1, wherein each of the graphics accelerators receives the same set of graphics primitives.

3. (Original) The graphics system of claim 1, wherein the dither values corresponding to the set of graphics accelerators have an average value of ½.

4. (Original) The graphics system of claim 1, wherein the dither values corresponding to the set of graphics accelerators have an average value of 2 to a power J, wherein J is an integer.

5. (Original) The graphics system of claim 1, wherein the dither values corresponding to the set of graphics accelerators have a dither radius greater than or equal to one.

6. (Currently Amended) The graphics system of claim 1, wherein a filtering unit calculates its corresponding partial sums from a set of the truncated versions of the dithered color components the filtering unit receives from a corresponding graphics accelerator, and wherein said set corresponds to sample locations that are within a filter support region for a location of the specific pixel.

7. (Previously Presented) The graphics system of claim 1, wherein the series of filtering units are configured to add the partial sums in a pipelined fashion.

8. (Original) The graphics system of claim 7, wherein a last of the filtering units in said series is configured to normalize a set of final cumulative sums resulting from said addition of the partial sums in a pipelined fashion.

9. (Previously Presented) The graphics system of claim 1, wherein each graphics accelerator comprises a plurality of sets of components, wherein each set comprises a rendering processor, an internal frame buffer, and a video data port, and wherein each video data port on the card couples to a corresponding filtering unit.

10. (Currently Amended) A graphics System comprising:

a set of rendering processors, wherein each rendering processor is connected to a video data output port; and

a series of filtering units, wherein each of the filtering units couples to a corresponding one of the video data output ports;

wherein each rendering processor RP(K) of the set of rendering processors is configured to:

(a) generate a stream of samples in response to received graphics primitives,

(b) add a dither value D_K to a data component of each of the samples in the stream to obtain dithered data components,

(c) buffer the dithered data components in an internal frame buffer, and

(d) forward a truncated version of the dithered data components to the corresponding filtering unit; [[and]]

wherein each of the filtering units is configured to compute for a specific pixel a partial sum of those dithered data components that are received from a corresponding rendering processor and that correspond to locations within a filter support region for the specific pixel location[[,]]; and

wherein the filtering units are configured to add their partial sums for the specific pixel in a pipelined fashion.

11. (Original) The graphics system of claim 10, wherein the rendering processors reside within original equipment manufacturer (OEM) graphics cards.

12. (Previously Presented) The graphics system of claim 11, wherein each of the graphics cards contains two of the rendering processors and two video data output ports, and wherein each video data output port is connected to a different one of the rendering processors.

13. (Original) The graphics system of claim 10, wherein the sample data component is a color component.

14. (Previously Presented) The graphics system of claim 10, wherein a data component is an alpha component.

15. (Original) The graphics system of claim 10, wherein the dither values corresponding to the set of graphics accelerators have an average value of 2 to a power J, wherein J is an integer.

16. (Canceled)

17. (Previously Presented) The graphics system of claim 10, wherein a last of the filtering units in said series is configured to normalize a set of final sums resulting from said addition of the partial sums in a pipelined fashion.

18. (Previously Presented) A method comprising:

broadcasting a stream of graphics primitives to a plurality of rendering processors;

each rendering processor RP(K) of said plurality of rendering processors:

(a) generating a stream of samples in response to received graphics primitives,

(b) adding a dither value D_K to a data component of each of the samples in the stream to obtain dithered data components,

(c) buffering the dithered data components in an internal frame buffer, and

(d) forwarding a truncated version of the dithered data components to a corresponding filtering unit of a series of filtering units; and

performing a weighted averaging computation in the series of filtering units in a pipelined fashion on the truncated dithered data components to determine data components for a pixel, wherein each of the filtering units is configured to support the weighted averaging computation by computing a corresponding partial sum for those data components that correspond to samples that are located within a filter support region for the location of the pixel.

19. (Original) The method of claim 18, wherein the rendering processors reside within a set of original equipment manufacturer (OEM) graphics cards.

20. (Previously Presented) The method of claim 19, wherein each of the graphics cards contains one or more of the rendering processors.

21. (Original) The method of claim 18, wherein the data component is a color component.

22. (Original) The method of claim 18, wherein the data component is an alpha component.

23. (Original) The method of claim 18, wherein the dither values corresponding to the set of graphics accelerators have an average value of 2 to a power J, wherein J is an integer.

24. (Previously Presented) The method of claim 18, wherein the series of filtering units are configured to add the partial sums in a pipelined fashion, and wherein a last of the filtering units in said series is configured to normalize a set of final cumulative sums resulting from said addition of the partial sums in a pipelined fashion.

25-26. (Canceled)